

LAPORAN PROYEK AKHIR KELOMPOK
KLASIFIKASI PENYAKIT GINJAL DARI CITRA CT SCAN GINJAL

Mata Kuliah Pencitraan Medis



Disusun Oleh
Kelompok 5

Faris Hassan Aly Firman
Fakhri Anargya Arrafi
Erfina Arthur
Anzilina Rahma

NPM. 1906381552
NPM. 2006578280
NPM. 2006465281
NPM. 2006578192

DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA

DESEMBER 2022

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmatNya, saya dapat menyelesaikan laporan proyek akhir ini. Penulisan laporan proyek akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Biomedik dan Teknik Komputer pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan laporan proyek akhir ini, sangatlah sulit bagi kami untuk menyelesaikan laporan proyek akhir ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Drs. Basari, selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan kami dalam penyusunan laporan proyek akhir ini;
- (2) Md Nazmul Islam dan Md Humaion Kabir Mehedi yang telah banyak membantu dalam usaha memperoleh dataset yang kami perlukan;
- (3) Akshay Radhakrishnan yang telah menyediakan source code yang kami perlukan;
- (4) Orang tua dan keluarga kami yang telah memberikan bantuan dukungan material dan moral; dan
- (5) Sahabat yang telah banyak membantu kami dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga laporan proyek akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, 28 Desember 2022
Kelompok 5

ABSTRAK

Penyakit ginjal adalah kondisi serius dan berpotensi mengancam jiwa yang mempengaruhi jutaan orang di seluruh dunia. Diagnosis dini dan pengobatan penyakit ginjal dapat meningkatkan hasil dan kualitas hidup pasien secara signifikan. Dalam penelitian ini, penulis mengusulkan sebuah metode untuk mengklasifikasikan penyakit ginjal menggunakan deep learning convolutional neural network (CNN) yang diterapkan pada gambar pemindaian computed tomography (CT) ginjal. Pada langkah pra pemrosesan akan melibatkan teknik seperti memangkas dan mengubah ukuran gambar ke ukuran yang konsisten, serta mengoreksi setiap bias atau distorsi pada gambar. Kemudian digunakan model CNN yang dilatih pada kumpulan data besar gambar CT scan untuk mengklasifikasikan gambar ke dalam salah satu dari beberapa kategori penyakit ginjal. Hasil penelitian menunjukkan bahwa model CNN mampu mengklasifikasikan penyakit ginjal secara akurat dengan tingkat akurasi yang tinggi, menunjukkan potensi teknik pembelajaran mendalam untuk diagnosis dini dan pengobatan penyakit ginjal. Penelitian ini menyajikan pendekatan yang menjanjikan untuk klasifikasi penyakit ginjal menggunakan CNN pembelajaran mendalam yang diterapkan pada gambar CT scan. Tingkat akurasi yang tinggi yang dicapai oleh model CNN menunjukkan bahwa pendekatan ini dapat menjadi alat yang berharga untuk meningkatkan hasil dan kualitas hidup pasien bagi mereka yang menderita penyakit ginjal. Penelitian lebih lanjut diperlukan untuk menentukan generalisasi hasil ini dan untuk mengeksplorasi penggunaan pendekatan ini dalam pengaturan klinis.

Kata kunci:

Penyakit ginjal, Klasifikasi ginjal, Convolutional Neural Network, CT SCAN

Abstract

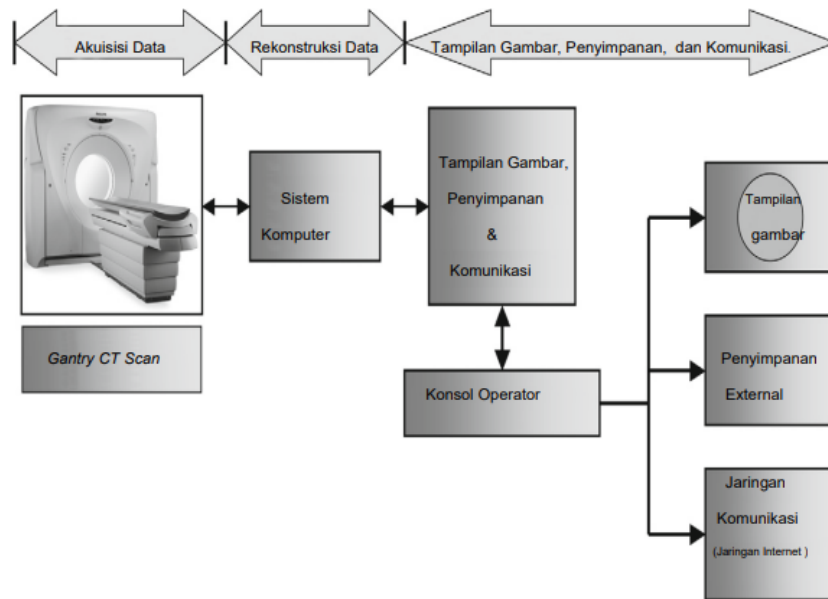
Kidney disease is a serious and potentially life-threatening condition that affects millions of people worldwide. Early diagnosis and treatment of kidney disease can significantly improve patient outcomes and quality of life. In this study, the authors propose a method for classifying kidney disease using a deep learning convolutional neural network (CNN) applied to computed tomography (CT) scan images of the kidneys. The pre-processing step will involve techniques such as cropping and resizing the image to a consistent size, as well as correcting any bias or distortion in the image. Then we used the CNN model trained on a large data set of CT scan images to classify the images into one of several categories of kidney disease. The results show that the CNN model is able to accurately classify kidney disease with a high degree of accuracy, demonstrating the potential of deep learning techniques for early diagnosis and treatment of kidney disease. This study presents a promising approach for classification of kidney disease using deep learning CNN applied to CT scan images. The high degree of accuracy achieved by the CNN model suggests that this approach can be a valuable tool for improving patient outcomes and quality of life for those with kidney disease. Further research is needed to determine the generalizability of these results and to explore the use of this approach in clinical settings.

Keyword: Kidney disease, Classification of kidney, Convolutional Neural Network, CT SCAN

DAFTAR ISI

KLASIFIKASI PENYAKIT GINJAL DARI CITRA CT SCAN GINJAL	1
KATA PENGANTAR	2
ABSTRAK	3
DAFTAR ISI	4
DAFTAR GAMBAR	6
DAFTAR TABEL	6
BAB 1	8
PENDAHULUAN	8
1.1. Latar Belakang	8
1.2 Rumusan Masalah	9
1.3 Tujuan Laporan	9
1.4 Batasan Masalah	9
1.5 Sistematika Laporan	10
BAB 2	11
TINJAUAN PUSTAKA	11
2.1 Pencitraan Computed Tomography	11
2.2 Batu Ginjal	12
2.3 Tumor Ginjal	12
2.4 Kista Ginjal	12
2.5 Metode Komputasi dalam Deteksi dan Prediksi Abnormalitas pada Ginjal	13
2.5.1 Pre-Processing	13
2.5.2 Machine Learning	13
BAB 3	14
METODOLOGI	14
3.1 Alat dan Bahan	14
3.2 Metode yang dipakai	14
3.3 Hasil Simulasi dan Pembahasan	15
BAB 4	17
PENUTUP	17
4.1 Kesimpulan	17
DAFTAR PUSTAKA	18

DAFTAR GAMBAR



Gambar 2.1. Komponen dan proses pada alat CT Scan

```

In [29]: pic=[]
img = cv2.imread(str('archive/CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone/Cyst/Cyst- (1004).jpg'))
img = cv2.resize(img, (28,28))
if img.shape[2] == 1:
    img = np.dstack([img, img, img])
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
img=np.array(img)
img = img/255
#label = to_categorical(0, num_classes=2)
pic.append(img)
#pic_labels.append(pneu)

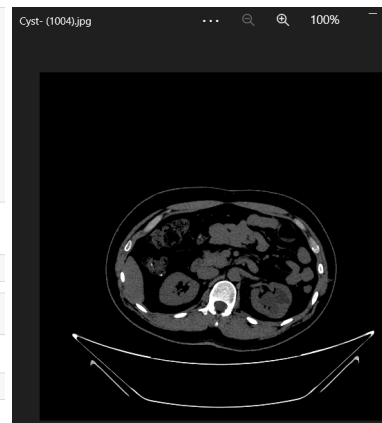
im

In [30]: pic1 = np.array(pic)

In [31]: a=model.predict(pic1)
a
Out[31]: array([[ 8.246049 , -8.904935 , -7.2466297, -6.8986206]], dtype=float32)

In [32]: a.argmax()
Out[32]: 0

```



Gambar 3.1 [a.(kiri) bagian koding simulasi, b.(kanan) gambar CT scan ginjal dengan kista]

DAFTAR TABEL

Alat	Keterangan
Komputer/Laptop	Milik Pribadi

Python3/Google Collab/Kaggle Workspace	Akses Gratis
Code Editor	Akses Gratis

Tabel 3.1. Daftar alat yang digunakan kelompok.

Bahan	Keterangan
<i>Source Code 'KIDNEY-diseases 0.999 accuracy'</i>	Dibuat oleh Akshay Radhakrishnan , diakses melalui Kaggle.com
<i>Dataset 'CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone'</i>	Diupload oleh Md Nazmul Islam dan Md Humaion Kabir Mehedi , diakses melalui Kaggle.com

Tabel 3.2. Daftar alat yang digunakan kelompok.

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Penyakit ginjal merupakan masalah kesehatan yang serius dan semakin lazim di seluruh dunia, mempengaruhi jutaan orang dari segala usia. Hal ini ditandai dengan hilangnya fungsi ginjal secara progresif, yang dapat menyebabkan berbagai komplikasi dan peningkatan morbiditas dan mortalitas.

Ada beberapa penyebab penyakit ginjal, termasuk diabetes, tekanan darah tinggi, dan kondisi bawaan. Seringkali penderita tanpa gejala pada tahap awal, membuat deteksi dini dan diagnosis penting untuk pengobatan dan manajemen yang efektif. Namun, metode tradisional untuk mendeteksi dan mendiagnosis penyakit ginjal, seperti tes darah dan urine, mungkin tidak mendeteksi penyakit sampai kerusakan yang signifikan telah terjadi.

Salah satu metode umum untuk mendeteksi dan mendiagnosis penyakit ginjal adalah melalui penggunaan teknik pencitraan medis, seperti pemindaian tomografi komputer (CT). CT scan memberikan gambar rinci ginjal dan jaringan di sekitarnya, memungkinkan dokter untuk mengidentifikasi kelainan dan menilai tingkat kerusakan. Namun, menginterpretasikan CT scan secara manual dapat memakan waktu dan subjektif, dan mungkin tidak selalu dapat diandalkan.

Kecerdasan buatan (AI) berpotensi meningkatkan akurasi dan efisiensi diagnosis penyakit ginjal dengan mengotomatiskan analisis CT scan. Algoritma pembelajaran mesin dapat dilatih untuk mengenali pola dan fitur dalam gambar CT scan yang mengindikasikan penyakit ginjal, dan untuk mengklasifikasikan gambar sebagai sehat atau sakit dengan akurasi tinggi.

Ada beberapa penelitian sebelumnya tentang penggunaan AI untuk diagnosis penyakit ginjal dari gambar CT scan. Namun, sebagian besar studi ini berfokus pada klasifikasi biner (yaitu, keadaan sehat vs. keadaan sakit) atau menggunakan kumpulan data yang relatif kecil. Selain itu, ada kurangnya konsensus tentang algoritma pembelajaran mesin terbaik dan fitur gambar yang akan digunakan untuk tugas ini, serta tentang utilitas klinis dan batasan diagnosis penyakit ginjal berbasis AI.

Kami mengevaluasi pendekatan berbasis pembelajaran mesin untuk mengklasifikasikan penyakit ginjal dari citra CT scan ginjal. Kami akan menggunakan kumpulan data besar gambar beranotasi dan akan membandingkan ekstraksi fitur gambar. Tujuan utama kami adalah untuk mengembangkan model yang dapat secara akurat mengklasifikasikan penyakit

ginjal dari gambar CT scan dan memiliki potensi untuk digunakan secara klinis untuk diagnosis dan pengelolaan penyakit ginjal.

Selain mengevaluasi kinerja model, kami juga akan mengidentifikasi fitur dan pola paling penting dalam gambar CT scan yang mengindikasikan penyakit ginjal, dan akan mengeksplorasi bagaimana model menggunakan fitur ini untuk membuat prediksinya. Ini akan memberikan wawasan tentang mekanisme yang mendasari penyakit ginjal dan dapat membantu meningkatkan pemahaman kita tentang penyakit ini.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka penulis merumuskan masalah menjadi beberapa, diantaranya:

1. Metode apakah yang dapat digunakan untuk mengklasifikasikan penyakit ginjal secara akurat dari gambar CT scan?
2. Bagaimanakah proses kerja dari metode tersebut untuk dapat mengindikasi jenis penyakit ginjal yang dimiliki pasien?
3. Bagaimanakah hasil klasifikasi dari metode tersebut untuk dapat mengindikasi jenis penyakit ginjal yang dimiliki pasien?
4. Apa sajakah kelebihan dan kekurangan dari metode klasifikasi yang digunakan?

1.3 Tujuan Laporan

Tujuan dari penelitian ini adalah untuk mengevaluasi pendekatan berbasis pembelajaran mesin untuk mengklasifikasikan penyakit ginjal dari citra CT scan ginjal. Secara khusus, kami bertujuan untuk:

1. Mengembangkan model pembelajaran mesin yang dapat mengklasifikasikan penyakit ginjal secara akurat dari gambar CT scan, menggunakan kumpulan data besar gambar beranotasi.
2. Identifikasi fitur dan pola paling penting dalam gambar CT scan yang mengindikasikan penyakit ginjal, dan pahami bagaimana model menggunakan fitur ini untuk membuat prediksinya.
3. Mendalami kegunaan klinis potensial dari model ini dalam membantu mendiagnosis dan mengelola penyakit ginjal, dan mengidentifikasi batasan atau tantangan potensial dalam penggunaannya.

1.4 Batasan Masalah

Untuk membuat penyusun fokus pada satu bahasan, maka makalah ini dibatasi dengan:

1. Aplikasi pengklasifikasian penyakit ginjal diidentifikasi dari dataset berbagai rumah sakit di bangladesh dengan 15.000 data pasien
2. Metode yang diusulkan didasarkan pada algoritma pembelajaran mesin, yang peka terhadap kualitas dan kuantitas data yang digunakan untuk pelatihan dan pengujian.
3. Kemungkinan bahwa algoritma pembelajaran mesin yang digunakan dalam penelitian ini mungkin telah disesuaikan dengan data pelatihan dan mungkin tidak dapat digeneralisasikan dengan baik ke data baru yang tidak terlihat.

1.5 Sistematika Laporan

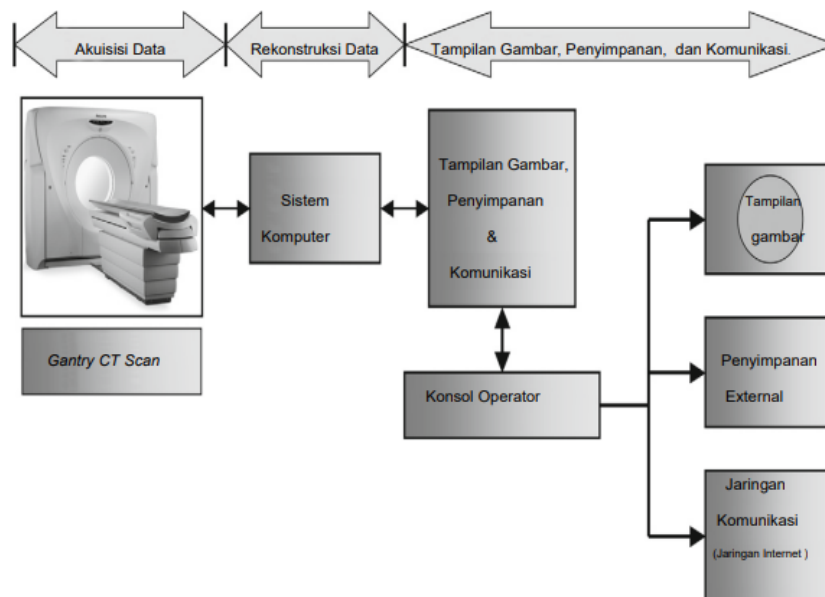
Laporan telah disusun berdasarkan hasil diskusi dan penelitian terkait klasifikasi gambar dari hasil CT scan pasien dengan keadaan ginjal sehat, pengidap kista ginjal, pengidap batu ginjal, dan pengidap tumor ginjal menggunakan deep learning dengan metode *Convolutional Neural Network* (CNN). Penyusunan laporan telah disesuaikan dengan aturan Pedoman Teknik Penulisan Tugas Akhir Mahasiswa Universitas Indoensia sesuai SK Rektor Universitas Indonesia dengan nomor 2143/SK/R/UI/2017. Mengenai lampiran tugas lainnya seperti *slides* dan video presentasi telah dilakukan pembagian tugas yang adil dan telah disepakati seluruh penulis. Secara keseluruhan, pengerjaan dan penyusunan tugas akhir mata kuliah Teknologi Pencitraan Medis telah dikerjakan bersama oleh seluruh penulis.

BAB 2

TINJAUAN PUSTAKA

2.1 Pencitraan Computed Tomography

Pemeriksaan CT Scan atau *Computed Tomography* Scan adalah salah satu modalitas dalam pencitraan medis yang bertujuan untuk mendapatkan gambar atau citra tubuh seperti jaringan lunak, tulang, pembuluh darah, dan lain-lain. Pemeriksaan CT Scan menggunakan mesin sinar-X yang dapat berputar untuk menghasilkan gambar yang kemudian digunakan untuk keperluan diagnosis penyakit. Cara pemeriksaan pada CT Scan hampir mirip dengan mesin MRI di mana pasien akan terbaring pada suatu tempat tidur kemudian mesin tersebut akan melakukan pemindaian. Perbedaan antara CT Scan dan MRI adalah alat pada CT Scan terbuka dan kemudian bergerak melalui lubang pemindai, sedangkan tempat tidur MRI masuk ke dalam ruang yang tertutup¹. Sumber yang digunakan dalam CT Scan adalah sinar X. Daerah lingkup pemeriksaan CT Scan biasanya pada daerah dada, perut, panggul, tungkai, tulang belakang, dan saluran kemih².



Gambar 2.1. Komponen dan proses pada alat CT Scan

Proses yang terdapat pada CT Scan antara lain proses akuisisi data, pengolahan data, rekonstruksi citra, representasi citra, penyimpanan citra, dan dokumentasi. Pada akuisisi data, akan diukur dan dikumpulkan transmisi sinar-X yang diterima pasien yang kemudian akan masuk ke dalam detektor untuk diukur nilai transmisi atau atenuasinya. Data-data yang sudah didapat dari proses akuisisi, akan digunakan pada proses rekonstruksi. Data yang diperoleh kemudian akan diubah menjadi sinyal listrik oleh detektor yang kemudian akan diubah lagi menjadi data digital melalui serangkaian pemrosesan oleh komputer. Komputer mengaplikasikan algoritma matematika tertentu yang digunakan dalam rekonstruksi,

misalnya pada alat pertama CT Scan digunakan algoritma Hounsfield. Setelah proses rekonstruksi, hasil gambar dapat ditampilkan dan disimpan yang nantinya dapat digunakan untuk keperluan diagnosis dengan melakukan serangkaian proses *pre-processing* terlebih dahulu³.

2.2 Batu Ginjal

Ginjal adalah salah satu organ penyusun tubuh manusia. Ginjal terletak di sisi kiri dan kanan sedikit belakang panggul. Ginjal berperan penting dalam membersihkan sisa-sisa bahan dari darah dengan membuat urin dan juga menyeimbangkan konsentrasi berbagai elemen dan mineral dalam tubuh. Selain itu, ginjal juga berfungsi untuk menyeimbangkan atau menyesuaikan pH, konsentrasi garam dan kalsium dalam tubuh. Gangguan pada ginjal menyebabkan ginjal tidak bisa menjalankan fungsinya secara optimal di mana salah satu gangguan atau penyakit yang menyerang pada ginjal adalah batu ginjal atau *kidney stone*.

Batu ginjal adalah kondisi di mana terdapatnya endapan padat di dalam ginjal yang berasal dari zat-zat yang terlarut dalam urin. Batu ginjal dapat terjadi di sepanjang saluran urin yaitu ureter, kandung kemih, dan uretra. Batu ginjal yang berukuran kecil biasanya tidak menimbulkan gangguan, tetapi seiring berjalannya waktu batu ginjal akan membesar dan semakin keras sehingga diperlukan suatu intervensi untuk mendeteksi keberadaan batu ginjal di mana salah satunya dengan menggunakan modalitas CT Scan.

2.3 Tumor Ginjal

Tumor atau kanker ginjal adalah kondisi di mana sel-sel ginjal tumbuh secara tidak terkendali. Kanker ginjal, seperti pada kanker umumnya, biasanya disebabkan oleh adanya mutasi pada sel-sel ginjal yang membuat sel tersebut tumbuh tidak normal dan berkembang membentuk suatu massa yang dikenal dengan tumor. Perbedaan kanker dan tumor adalah bila tumor tersebut memiliki kemampuan menyerang jaringan dan menyebar ke organ lain maka disebut dengan kanker. Sama halnya dengan batu ginjal, pada tumor atau kanker ginjal, pasien pada tahapan awal biasanya tidak mengalami gejala apapun. Apabila tumor semakin membesar, akan muncul gejala seperti keluarnya darah pada urin, nyeri pada punggung bagian bawah, dan lain-lain. Setelah munculnya gejala-gejala yang kerap terjadi, biasanya pasien akan melakukan konsultasi ke dokter dan kemudian dari dokter akan menyarankan pasien untuk menjalankan beberapa pemeriksaan dan salah satu pemeriksaan yang dianjurkan adalah CT Scan karena hasil gambar CT Scan yang jelas terhadap anatomi tubuh akan membantu dokter dalam mendiagnosis penyakit tumor atau kanker ginjal pada pasien.

2.4 Kista Ginjal

Kista ginjal merupakan sebuah penyakit pada ginjal akibat munculnya kantung berisi cairan (kista) di dalam jaringan ginjal. Kista ginjal dapat muncul pada salah satu atau kedua ginjal. Penyebab terbentuknya kista ginjal belum diketahui secara pasti, namun faktor usia diduga mempengaruhi kondisi ini. Kista ginjal biasanya jinak, tidak berbahaya, dan jarang

menimbulkan gejala. Selain itu, kista ginjal berbeda dengan penyakit ginjal polikistik yang disebabkan oleh faktor genetik.

2.5 Metode Komputasi dalam Deteksi dan Prediksi Abnormalitas pada Ginjal

2.5.1 Pre-Processing

Image processing adalah salah satu bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar yang nantinya akan dilakukan serangkaian proses sehingga input gambar tersebut akan mengeluarkan output yang nantinya dapat diolah oleh komputer. Sebelum dilakukan pengolahan gambar, akan dilakukan proses *pre-processing* di mana pada bagian ini akan dihilangkan bagian-bagian yang tidak diperlukan pada input gambar. Langkah *pre-processing* yang dilakukan dalam deteksi dan prediksi abnormalitas pada ginjal adalah dengan memberikan label pada setiap data dan menyeimbangkan jumlah data untuk setiap kelas penyakitnya. Metode tingkat-data atau yang dikenal dengan metode tingkat eksternal adalah suatu metode yang digunakan untuk memanipulasi data training yang digunakan. Metode ini digunakan untuk menyeimbangkan rasio sampel pada semua kelas dengan cara *undersampling* atau *oversampling*. Menghapus sebagian sampel dari kelas mayoritas maka disebut *undersampling*, sedangkan menghasilkan data buatan untuk kelas minoritas disebut dengan *oversampling*. Dengan dilakukannya penyerataan jumlah setiap kelas, akan diperoleh akurasi prediksi yang lebih representatif. Salah satu teknik yang merupakan turunan dari *oversampling* adalah SMOTE (Synthetic Minority Over-sampling Technique). Pada SMOTE, sampel baru akan disintesis dari kelas minoritas. Penerapan SMOTE pada bidang klasifikasi menunjukkan perbaikan performa dengan metode *Multi Level Perceptron* (MLP), *k-Nearest Neighbor* (k-NN), *Naive Bayes*, *Random Forest*, dan *Support Vector Machine*.

2.5.2 Machine Learning

Machine learning adalah suatu ilmu pengembangan algoritme yang digunakan pada sistem komputer untuk memproses data yang berjumlah besar dan mengidentifikasi pola data sehingga dapat dijalankan tugas tanpa instruksi yang eksplisit. Klasifikasi adalah salah satu topik dalam *data mining* atau *machine learning* di mana data awalnya dikelompokkan dengan diberi label atau target. Algoritma dalam menangani masalah klasifikasi dikelompokkan menjadi *supervised learning* atau pembelajaran yang diawasi dan *unsupervised learning* yang berarti tidak diawasi. *Supervised* di sini berarti data label ikut berperan sebagai *supervisor* yang mengawasi proses pembelajaran dalam mencapai tingkat akurasi tertentu, begitu pula sebaliknya untuk *unsupervised learning*. *Convolutional Neural Network* (CNN) adalah salah satu algoritma *machine learning* yang termasuk ke dalam *supervised learning* yang biasanya digunakan dalam menyelesaikan masalah regresi dan klasifikasi. Pada CNN, algoritma akan mengidentifikasi struktur patahan pada data visual atau

BAB 3

METODOLOGI

3.1 Alat dan Bahan

Alat yang digunakan untuk proyek akhir ini seluruhnya menggunakan milik pribadi dan juga memiliki izin untuk mempergunakannya, antara lain.

Alat	Keterangan
Komputer/Laptop	Milik Pribadi
Python3/Google Collab/Kaggle Workspace	Akses Gratis
Code Editor	Akses Gratis

Tabel 3.1. Daftar alat yang digunakan kelompok.

Sedangkan untuk bahan proyek dikumpulkan dari berbagai sumber di internet dengan rincian sebagai berikut.

Bahan	Keterangan
<i>Source Code 'KIDNEY-diseases 0.999 accuracy'</i>	Dibuat oleh Akshay Radhakrishnan , diakses melalui Kaggle.com
<i>Dataset 'CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone'</i>	Diupload oleh Md Nazmul Islam dan Md Humaion Kabir Mehedi , diakses melalui Kaggle.com

Tabel 3.2. Daftar alat yang digunakan kelompok.

3.2 Metode yang dipakai

Program 'KIDNEY-diseases 0.999 accuracy' yang kami gunakan merupakan deep learning yang menggunakan metode *Convolutional Neural Network* (CNN). CNN yang digunakan memiliki konfigurasi model sebagai berikut.

```
layers.Conv2D(28, (3, 3), activation='relu', input_shape=(28, 28, 3)) ,  
layers.MaxPooling2D((2, 2)),  
layers.Conv2D(64, (3, 3), activation='relu'),  
layers.MaxPooling2D((2, 2)),  
layers.Conv2D(64, (3, 3), activation='relu')
```

```

model.add(layers.Flatten())
model.add(layers.Dense(640, activation='tanh'))
model.add(layers.Dropout(0.5))
model.add(layers.Dense(264, activation='tanh'))
model.add(layers.Dense(64, activation='sigmoid'))

model.add(layers.Dense(4))

```

Tiga layer konvolusi dengan maxpooling dan juga fungsi aktivasi relu, kemudian flatten layer, dropout layer, dan empat dense layer dengan fungsi aktivasi tanh dan sigmoid.

Dataset yang digunakan merupakan *raw image* dari hasil dua belas ribu gambar CT Scan, sehingga terdapat beberapa proses preprocessing yang dilakukan seperti pemberian label di tiap gambar kemudian melakukan penyeimbangan terhadap jumlah dataset untuk tiap label menggunakan teknik SMOTE atau *Synthetic Minority Oversampling Technique* untuk mengatasi oversampling.

Secara *default*, penulis kode ini menggunakan epoch sebesar 100, namun kami melakukan penyesuaian dengan mengubahnya menjadi 20. Splitting data yang dilakukan menggunakan nilai 0.23 sehingga train data dipilih sebanyak 77% dan test data 23%.

3.3 Hasil Simulasi dan Pembahasan

Untuk mensimulasikan program, kita harus memasukan gambar yang kita ingin ujikan dengan mengubah direktori file pada bagian koding `img=cv2.imread(str(...))` seperti di gambar 1.a di bawah fungsi `pic=[]`. Contohnya, apabila kita memasukkan gambar CT scan ginjal dengan kista seperti yang di gambar 1.b, kita sesuaikan dengan direktorinya dan nama filenya yaitu `Cyst-(1004)` pada koding di gambar 1.a.

```

In [29]: pic=[]
img = cv2.imread(str('archive/CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone/Cyst/Cyst- (1004).jpg'))
img = cv2.resize(img, (28,28))
if img.shape[2] ==1:
    img = np.dstack([img, img, img])
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
img=np.array(img)
img = img/255
#label = to_categorical(0, num_classes=2)
pic.append(img)
#pic_labels.append(pneu)

im

In [30]: pic1 = np.array(pic)

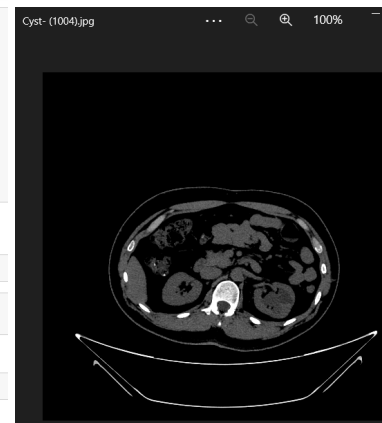
In [31]: a=model.predict(pic1)
a

Out[31]: array([[ 8.246049 , -8.904935 , -7.2466297, -6.8986206]], dtype=float32)

In [32]: a.argmax()

Out[32]: 0

```



Gambar 3.1 [a.(kiri) bagian koding simulasi, b.(kanan) gambar CT scan ginjal dengan kista]

Setelah itu, gambar akan diupload ke program dan di read oleh fungsi `cv2`. Kemudian, gambar akan dijadikan array yang akan discan oleh fungsi `model.predict()` yang kita dapatkan dari hasil train dan test komputer yang kita program sebelumnya berdasarkan metode CNN.

Untuk mendapatkan hasil akhir, kita menggunakan fungsi `argmax ()` dari library NumPy yang mengembalikan indeks elemen maksimum array sepanjang sumbu tertentu. Indeks elemen

yang didapat akan diidentifikasi sesuai dengan yang kita sudah tetapkan di awal kode, yaitu 0 untuk Kista, 1 untuk Normal, 2 untuk Batu, 3 untuk Tumor. Dapat dilihat pada gambar 1.a bahwa hasil `argmax()` nilainya adalah 0 yang merupakan kategori kista yang ternyata sesuai dengan datasetnya. Hal ini menandakan program ini berhasil untuk mengklasifikasi gambar batu ginjal, begitu juga setelah diujicobakan kepada sampel jenis ginjal lainnya.

BAB 4

PENUTUP

4.1 Kesimpulan

Program ini dilatih dengan dataset *CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone* yang di-publish oleh Md Nazmul Islam dan Md Humaion Kabir Mehedi pada website [Kaggle.com](https://www.kaggle.com). Program ini menggunakan deep learning dengan metode *Convolutional Neural Network* (CNN) untuk membuat klasifikasi empat gambar ginjal dan sumber kode yang kami gunakan adalah *KIDNEY-diseases 0.999 accuracy* yang dibuat oleh Akshay Radhakrishnan melalui website [Kaggle.com](https://www.kaggle.com).

Berdasarkan penyesuaian pada kode yang kami lakukan, hanya dibutuhkan epoch sebesar 20 untuk mendapatkan akurasi klasifikasi sebesar 0.999. Program ini dapat berjalan sesuai fungsinya untuk mengklasifikasi berbagai gambar CT scan ginjal dengan akurasi sebesar 99%. Sehingga dapat disimpulkan bahwa program ini secara akurat dapat mengklasifikasi empat jenis kondisi ginjal, yaitu ginjal normal, ginjal dengan kista, ginjal dengan tumor, dan ginjal dengan batu ginjal hanya dengan menginput hasil CT scan ginjal.

DAFTAR PUSTAKA

- Ali, H. (2019). A review on data preprocessing methods for class imbalance problem. *10(10449)*, 390-397.
https://www.researchgate.net/publication/336305785_A_review_on_data_preprocessing_methods_for_class_imbalance_problem/citations
- Batu Ginjal - Gejala, penyebab dan mengobati.* (2022, March 28). Alodokter. Retrieved December 28, 2022, from <https://www.alodokter.com/batu-ginjal>
- Islam, N., & Mehedi, H. K. (2021). *CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone*. Kaggle. Retrieved December 28, 2022, from <https://www.kaggle.com/datasets/nazmul0087/ct-kidney-dataset-normal-cyst-tumor-and-stone>
- Jangan Keliru Lagi, Ini Perbedaan CT Scan dan MRI.* (2021, May 29). KlikDokter. Retrieved December 28, 2022, from <https://www.klikdokter.com/info-sehat/kesehatan-umum/jangan-keliru-lagi-ini-perbedaan-ct-scan-dan-mri>
- Kamu Harus Tahu Perbedaan CT Scan dan MRI Kamu Harus Tahu Perbedaan CT Scan dan MRI.* (n.d.). Primaya Hospital. Retrieved December 28, 2022, from <https://primayahospital.com/radiologi/perbedaan-ct-scan-dan-mri/>
- Kanker Ginjal - Gejala, Penyebab, dan Pengobatan.* (n.d.). Halodoc. Retrieved December 28, 2022, from <https://www.halodoc.com/kesehatan/kanker-ginjal>
- Marena, M. (2021, May 10). *CT Scan, Ini yang Harus Anda Ketahui.* Alodokter. Retrieved December 28, 2022, from <https://www.alodokter.com/ct-scan-ini-yang-harus-anda-ketahui>
- Radhakrishnan, A. (2022). *KIDNEY-diseases 0.999 accuracy*. Kaggle. Retrieved December 28, 2022, from <https://www.kaggle.com/code/akshayr009/kidney-diseases-0-999-accuracy>
- Siringoringo, R. (2018, January 1). *KLASIFIKASI DATA TIDAK SEIMBANG MENGGUNAKAN ALGORITMA SMOTE DAN k-NEAREST NEIGHBOR.* Retrieved

December 28, 2022, from

<https://ejournal-medan.uph.edu/index.php/isd/article/viewFile/177/63>

10 BAB II TINJAUAN PUSTAKA A. Tinjauan Teori 1. Prinsip Dasar CT Scan (Seeram E.,

2009) Prinsip fisika yang terdapat pada CT Sca. (n.d.). Repository Poltekkes

Semarang. Retrieved December 28, 2022, from

<https://repository.poltekkes-smg.ac.id/repository/BAB%20II%20AFIF%20P1337430218087.pdf>

What is Machine Learning? - Enterprise Machine Learning Beginner's Guide. (n.d.). AWS.

Retrieved December 28, 2022, from

<https://aws.amazon.com/id/what-is/machine-learning/>

LAMPIRAN

Source Code

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import os
import cv2
from pathlib import Path
import seaborn as sns
import matplotlib.pyplot as plt
from skimage.io import imread

# Define path to the data directory
data_dir =
Path('../input/ct-kidney-dataset-normal-cyst-tumor-and-stone/CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone/')

# Path to train directory
train_dir = data_dir / 'CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone'
train_dir

# Get the path to the normal and pneumonia sub-directories
normal_cases_dir = train_dir / 'Normal'
Cyst_cases_dir = train_dir / 'Cyst'
Stone_cases_dir = train_dir / 'Stone'
```

```

Tumor_cases_dir = train_dir / 'Tumor'

# Get the list of all the images
normal_cases = normal_cases_dir.glob('*.jpg')
Cyst_cases = Cyst_cases_dir.glob('*.jpg')
Stone_cases = Stone_cases_dir.glob('*.jpg')
Tumor_cases = Tumor_cases_dir.glob('*.jpg')

# An empty list. We will insert the data into this list in (img_path, label) format
train_data = []

# Go through all the Cyst_cases . The label for these cases will be 1
for img in Cyst_cases:
    train_data.append((img, 0))
    # Go through all the normal cases. The label for these cases will be 0
for img in normal_cases:
    train_data.append((img, 1))

# Go through all the normal cases. The label for these cases will be 0
for img in Stone_cases:
    train_data.append((img, 2))

# Go through all the Tumor_cases . The label for these cases will be 1
for img in Tumor_cases:
    train_data.append((img, 3))

# Get a pandas dataframe from the data we have in our list
train_data = pd.DataFrame(train_data, columns=['image', 'label'], index=None)

# Shuffle the data
train_data = train_data.sample(frac=1.).reset_index(drop=True)

# How the dataframe looks like?
train_data.head()

train_data['label'].unique()

train_data.shape

# Get the counts for each class
cases_count = train_data['label'].value_counts()
print(cases_count)

```

```

# Plot the results
plt.figure(figsize=(10,8))
sns.barplot(x=cases_count.index, y=cases_count.values)
plt.title('Number of cases', fontsize=14)
plt.xlabel('Case type', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.xticks(range(len(cases_count.index)), ['Cyst(0)', 'Normal(1)', 'Stone(2)', 'Tumor(3)'])
plt.show()

# Get few samples for both the classes
Cyst_samples = (train_data[train_data['label']==0]['image'].iloc[:5]).tolist()
Normal_samples = (train_data[train_data['label']==1]['image'].iloc[:5]).tolist()
Stone_samples = (train_data[train_data['label']==2]['image'].iloc[:5]).tolist()
Tumor_samples = (train_data[train_data['label']==3]['image'].iloc[:5]).tolist()

# Concat the data in a single list and del the above two list
samples = Cyst_samples + Normal_samples + Stone_samples + Tumor_samples
del Cyst_samples, Normal_samples, Stone_samples, Tumor_samples

# Plot the data
f, ax = plt.subplots(4,5, figsize=(30,30))
for i in range(20):
    img = imread(samples[i])
    ax[i//5, i%5].imshow(img, cmap='gray')
    if i<5:
        ax[i//5, i%5].set_title("Cyst_samples")
    elif i<10:
        ax[i//5, i%5].set_title("Normal_samples")
    elif i<15:
        ax[i//5, i%5].set_title("Stone_samples")
    elif i<20:
        ax[i//5, i%5].set_title("Tumor_samples")

    ax[i//5, i%5].axis('off')
    ax[i//5, i%5].set_aspect('auto')
plt.show()

# Get the path to the normal and pneumonia sub-directories
normal_cases_dir = train_dir / 'Normal'
Cyst_cases_dir = train_dir / 'Cyst'
Stone_cases_dir = train_dir / 'Stone'
Tumor_cases_dir = train_dir / 'Tumor'

```

```

# Get the list of all the images
normal_cases = normal_cases_dir.glob('*.jpg')
Cyst_cases = Cyst_cases_dir.glob('*.jpg')
Stone_cases = Stone_cases_dir.glob('*.jpg')
Tumor_cases = Tumor_cases_dir.glob('*.jpg')
train_data = []
train_labels = []
#from keras.utils.np_utils import to_categorical

for img in Cyst_cases:
    img = cv2.imread(str(img))
    img = cv2.resize(img, (28,28))
    if img.shape[2] ==1:
        img = np.dstack([img, img, img])
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img=np.array(img)
    img = img/255
    label = 'Cyst'
    train_data.append(img)
    train_labels.append(label)

for img in normal_cases:
    img = cv2.imread(str(img))
    img = cv2.resize(img, (28,28))
    if img.shape[2] ==1:
        img = np.dstack([img, img, img])
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img=np.array(img)
    img = img/255
    label = 'Normal'
    train_data.append(img)
    train_labels.append(label)

for img in Stone_cases:
    img = cv2.imread(str(img))
    img = cv2.resize(img, (28,28))
    if img.shape[2] ==1:
        img = np.dstack([img, img, img])
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img=np.array(img)
    img = img/255
    label = 'Stone'
    train_data.append(img)

```

```

train_labels.append(label)

for img in Tumor_cases:
    img = cv2.imread(str(img))
    img = cv2.resize(img, (28,28))
    if img.shape[2] == 1:
        img = np.dstack([img, img, img])
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img = np.array(img)
    img = img/255
    label = 'Tumor'
    train_data.append(img)
    train_labels.append(label)
# Convert the list into numpy arrays

train_data1 = np.array(train_data)
train_labels1 = np.array(train_labels)

print("Total number of validation examples: ", train_data1.shape)
print("Total number of labels:", train_labels1.shape)

train_data1.shape

train_data1[1]

train_labels1 = pd.DataFrame(train_labels1, columns=[ 'label'],index=None)
train_labels1.head()

train_labels1['label'].unique()

train_labels1['label']=train_labels1['label'].map({'Cyst':0,'Normal':1, 'Stone':2,'Tumor':3})

train_labels1

print(train_data1.shape)
print(train_labels1.shape)

train_labels1.isnull().sum()

from imblearn.over_sampling import SMOTE
smt = SMOTE()
train_rows=len(train_data1)
train_data1 = train_data1.reshape(train_rows,-1)
train_data2, train_labels2 = smt.fit_resample(train_data1, train_labels1)

```

```

cases_count1 = train_labels2['label'].value_counts()
print(cases_count1)

# Plot the results
plt.figure(figsize=(10,8))
sns.barplot(x=cases_count1.index, y= cases_count1.values)
plt.title('Number of cases', fontsize=14)
plt.xlabel('Case type', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.xticks(range(len(cases_count.index)), ['Cyst(0)', 'Normal(1)', 'Stone(2)', 'Tumor(3)'])
plt.show()

train_data2.shape

train_labels2.shape

train_labels2['label'].value_counts()

train_data2 =train_data2.reshape(-1,28,28,3)

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(train_data2, train_labels2, test_size=0.23,
random_state=42)

import tensorflow as tf
data_augmentation=tf.keras.preprocessing.image.ImageDataGenerator(
    rotation_range=30, horizontal_flip=True)

import tensorflow as tf
from tensorflow.keras import layers, models

model = models.Sequential([
    #data_augmentation,
    layers.Conv2D(28, (3, 3), activation='relu', input_shape=(28, 28, 3)) ,
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu')
])

model.summary()

```

```

model.add(layers.Flatten())
model.add(layers.Dense(640, activation='tanh'))
model.add(layers.Dropout(0.5))
model.add(layers.Dense(264, activation='tanh'))
model.add(layers.Dense(64, activation='sigmoid'))

model.add(layers.Dense(4))
model.summary()

model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])

model.fit(np.array(X_train), np.array(y_train), epochs=20, validation_data=(np.array(X_test),
np.array(y_test)))

pic=[]
img =
cv2.imread(str('../input/ct-kidney-dataset-normal-cyst-tumor-and-stone/CT-KIDNEY-DATAS
ET-Normal-Cyst-Tumor-Stone/CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone/Cyst/Cys
t- (1004).jpg'))
img = cv2.resize(img, (28,28))
if img.shape[2] ==1:
    img = np.dstack([img, img, img])
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
img=np.array(img)
img = img/255
#label = to_categorical(0, num_classes=2)
pic.append(img)
#pic_labels.append(pneu)

pic1 = np.array(pic)
a=model.predict(pic1)
a
a.argmax()

```